Simulating ISIS moderators with Mcstas

1 Introduction

The following document describes the functions obtained for models of TS2 as described in table 1:

target	3.4cm diameter tantalum clad tungsten
reflector	Be + D_2O (80:20) at 300K
Composite Moderator Coupled	H_2 + CH_4 Groove: 3x8.$\dot{3}$ cm 26K solid-CH_4 Hydrogen: 12x11cm 22K liquid H_2
Poisoned Moderator Decoupled	solid-CH_4 26K Narrow: Gd poison at 2.4 cm - 8 vanes Broad: 3.3 cm – not fully decoupled
PreModerators	0.85 cm and 0.75 cm H_2O

Table 1: **Description of Models**

TS1 model is from the tungsten target as currently installed and positioned. The model also includes the MERLIN moderator, this makes no significant difference to the other moderator faces.

2 Installing the McStas components

Version 0.8 of the ISIS moderator component for McStas has been distributed with McStas 1.8 and the following instructions are mainly for upgrading the tablefiles or an out-of-date component. The McStas components are contained in the compressed file ISISMOD.zip or ISISMOD.tgz, which can be unpacked using Winzip or

tar zxvf ISISMOD.tgz or
unzip ISISMOD.zip

this will create a directory ./ISIS_tables and unpack the ISIS Mcstas module 'ISIS_moderator.comp' and an example instrument file 'ISIStest.instr'. Copy or move the ISIS_moderator.comp to /mcstas/lib/contrib (or somewhere where mcstas will see it) and move the ./ISIS_tables directory to a place of your own convenience such as the default: '/usr/local/lib/mcstas/contrib/' or 'C:\mcstas\lib\contrib\'.

You MUST then set the environment variable 'MCTABLES' to be the full path of the directory containing the table files:

BASH: export MCTABLES=/usr/local/lib/mcstas/contrib/ISIS_tables/
TCSH: setenv MCTABLES /usr/local/lib/mcstas/contrib/ISIS_tables/

In Windows this can be done using the 'My Computer' properties and selecting the 'Advanced' tab and the Environment variables button.

3 Using the McStas Module

In order to test that the simulation has been correctly installed, an example instrument has been included (ISIStest.instr). The ISIS_moderator.comp program allows the inclusion of a primary component (ie it does not accept incoming neutrons). It requires a set of variables listed in table 1 and described below.

The *Face* variable determines the moderator surface that will be viewed. There are two types of *Face* variable: i) Views from the centre of each moderator face defined by the name of the moderator, for TS1: Water, H2, CH4, Merlin and TS2: Hydrogen, Groove, Narrow, Broad. ii) Views seen by each beamline, currently only available for TS2 E1-E9 (East) and W1-W9 (West).

Variables *E0* and *E1* define an energy window for sampled neutrons. This can be used to increase the statistical accuracy of chopper and mirrored instruments. However, *E0* and *E1* cannot be equal (although they can be close). By default these arguments select energy in meV, if negative values are given, selection will be in terms of Angstroms.

Variables *dist*, *xw* and *yh* are the three component which will determine the directional acceptance window. They define a rectangle with centre at (0,0,dist) from the moderator position and with width *xw* meters and height *yh* meters. The initial direction of all the neutrons are chosen (randomly) to originate from a point on the moderator surface and along a vector, such that without obstruction (and gravitational effects), they would pass through the rectangle. This should be used as a directional guide. All the neutrons start from the surface of the moderator and will be diverted/absorbed if they encountered other components. The guide system can be turned off by setting *dist* to zero.

The *CAngle* variable is used to rotate the viewed direction of the moderator and reduces the effective solid angle of the moderator face. Currently it is only for the horizontal plane.

The two variables *modYsize* and *modXsize* allow the moderators to be effectively reduced/increased. If these variables are given negative or zero values then they default to the actual visible surface size of the moderators.

The last variable *SAC* will correct for the different solid angle seen by two focussing windows which are at different distances from the moderator surface. The normal measurement of flux is in neutrons/second/Å/cm^2/str, but in a detector it is measured in neutrons/second. Therefore if all other denominators in the flux are multiplied out then the flux at a point-sized focus window should follow an inverse square law. This solid angle correction is made if the *SAC* variable is set equal to 1, it will not be calculated if *SAC* is set to zero. Provided simulations are made without changing the distance of the focussing window the correction need not be made.

Variable	Type	Options	Units	Description
Face (TS2)	char*	Hydrogen Groove Narrow Broad E1-E9 W1-W9	–	String which designates the name of the face
Face (TS1)	char*	H2 CH4 Merlin Water	–	String which designates the name of the face
E0	float	0<E0<E1	meV (Å)	Only neutrons above this energy are sampled
E1	float	E0<E1<1e10	meV (Å)	Only neutrons below this energy are sampled
dist	float	$0 < dist < \infty$	m	Distance of focus window from face of moderator
xw	float	$0 < xw < \infty$	m	x width of the focus window
yh	float	$0 < yh < \infty$	m	y height of the focus window
CAngle	float	-360 $< CAngle <$ 360	^{o}	Horizontal angle from the normal to the moderator surface
modXsize	float	$0 < modXsize < \infty$	m	Horizontal size of the moderator (defaults to actual size)
modYsize	float	$0 < modYsize < \infty$	m	Vertical size of the moderator (defaults to actual size)

4 Runtime output

Like many McStas components the runtime output is directed to the *stderr* channel (use mcprog.exe > logfile to save the output). During the initialisation of the component before events are generated, the chosen face, scale factor and selected energy channels are listed to this channel. During event production the only output is for unusual behaviour. If you experience any output from the component in event production, please send a bug report. Occasionally, you will see a bin-boundary error. This results is a resampling of the point and is not a problem unless the failed points are a significant number of the Ncount.

5 Test example

The test example contains just one moderator and it is positioned at the origin. A time and an energy detector are placed at 1.0 and 1.5 metres respectively from the moderator surface. They are positioned to see 100% of the neutrons passing through the focusing window at 1 metre. The program is compiled and run in the normal way for McStas. The results show the time and energy distributions of the neutrons.